descent

Minimize "energy" via a differential flow.

[called by: notopt.] [calls: knotxx.]

contents

1.1 overview

1. The geometry of the coils is "evolved" under the "energy gradient flow", where the energy function, $E(\mathbf{x})$, defined below, is considered to be a function of the coil geometry, $\mathbf{x} = \{x^i_{c,m}, x^i_{s,m}, y^i_{c,m}, y^i_{s,m}, z^i_{c,m}, z^i_{s,m}\}$, where the $x^i_{c,m}$ etc. are the Fourier harmonics of the i-th coil. The Fourier representation of the coils is described in iccoil.

2. The evolution is described mathematically as a system of coupled, first-order equations:
$$\frac{\partial \mathbf{x}}{\partial \tau} = -\frac{\partial E}{\partial \mathbf{x}}, \tag{1}$$
where τ is an artifical time.

3. The integration is performed using NAG:D02BJF, and is controlled by `tauend`, `tautol` and `Ntauout`.

1.2 derivarives test

1. When `Loptimize = -1`, the code will test the first derivatives comparing the results of finite difference, shudson's and czhu's method. Here, minus sign in shudson's method was temporarily eliminated.

2. When `Loptimize = -2`, the code will test the second derivatives comparing the results of finite difference and czhu's method. The results of finite difference come from using the first derivatives of shudson's method differentiating on the intervals. That is,
$$\frac{\partial^2 E}{\partial x_1 \partial x_2} \equiv \frac{\partial F_1}{\partial x_2} \equiv \frac{F_1(x_2 + \frac{1}{2}\Delta) - F_1(x_2 - \frac{1}{2}\Delta)}{\Delta} \tag{2}$$
Right now, the results of $\frac{\partial^2 E}{\partial {I_i}^2}$ don't match. But the results of czhu's method and finite difference result from 0-order functions, which is $\frac{\partial^2 E}{\partial {I_i}^2} \approx \frac{E(I_i - \frac{1}{2}\Delta) - 2E(I_i) + E(I_i - \frac{1}{2}\Delta)}{\Delta^2}$, do match.

3. *The second derivatives of energy function E on which variable were decided by the value jcoil and jdof, representing the coil number and the jdof-th DoF in each coil. Be careful with the differenced of DoF array arrangements between czhu's method and shudson's method.*

3. When `Loptimize = 1`, the code will perform differential flow method using NAG:D02BJF, with shudson's method to get the first derivatives. It's really fast and you can turn on/off Io, Lo to control the constrains of currents and length.

4. When `Loptimize = 2`, the code will perform differential flow method using NAG:D02BJF, with czhu's method to get the first derivatives. It's not so fast, but it can calculate the second derivatives. Like shudson's method, you can turn on/off the constrains of bnormal, toroidal flux and length (and more inthe later) through changing the value of `weight_bnorm`, `weight_tflux` and `weight_ttlen`.